Localization Algorithm for Wireless Ad-Hoc Sensor Networks with Traffic Overhead Minimization by Emission Inhibition

Leone, Pierre; Moraru, Luminita; Powell, Olivier; Rolim, Jose

How to cite

LEONE, Pierre et al. Localization Algorithm for Wireless Ad-Hoc Sensor Networks with Traffic Overhead Minimization by Emission Inhibition. In: Proceedings of the Second International Workshop on Algorithmic Aspects of Wireless Sensor Networks, ALGOSENSORS 2006. Venice (Italy). Berlin, Heidelberg : Springer, 2006. p. 119–129. (Lecture Notes in Computer Science) doi: 10.1007/11963271_11

This publication URL: https://archive-ouverte.unige.ch//unige:32656
Publication DOI: 10.1007/11963271_11

Localization algorithm for wireless ad-hoc sensor networks with traffic overhead minimization by emission inhibition

Pierre Leone, Luminita Moraru [*], Olivier Powell [**], and Jose Rolim

Department of Informatics,
University of Geneva, Switzerland
{leone, moraru, powell, rolim}@cui.unige.ch

Abstract. Widely used positioning systems like GPS are not a valid solution in large networks with small size, low cost sensors, due both to their size and their cost. Thus, new solutions for localization awareness are emerging, commonly based on the existence of a few references spread into the network.

We propose a localization algorithm to reduce the number of transmitting nodes. The algorithm relies on self selecting nodes for location information disclosure. Each node makes a decision based on its proximity to the nodes in the area covered only by two of the references used for its own localization. We analyze different aspects of the location awareness propagation problem: communication overhead, redundant transmissions, network coverage.

1 Introduction

The knowledge of the geographical position is necessary in a variety of sensor networks applications and communication protocols. In a sensor network, each entity uses its sensing capabilities to collect information about the surrounding area and to send it periodically or event driven to a base station, in a hop by hop manner. Routing protocols either use location as criteria for building hierarchical architectures for data dissemination [1] or to identify the area to be monitored for a specific event [2].

Although accurate localization techniques, like GPS devices are currently available, their usage for networks with a large number of sensors

---

[*] Research partially founded by the Swiss SER Contract No. C05.0030 and FP6-015964 AEOLUS

[**] Research partially supported by the Swiss National Science Foundation grant no. 20021-1040107

is difficult due to the cost and the size of the necessary equipment. The alternative is to use a small number of GPS enabled nodes, considered as references by the rest of the nodes of the network to localize themselves.

Current algorithms consider the aspect of accuracy and its impact on localization aware applications. One of the most important issues in sensor networks is the network lifetime. Since the main energy consumption source is represented by the communication mechanism [3], [4], [5], one of the main concern of any algorithm that require network traffic is to minimize the number of messages sent into the network.

We propose a method of minimizing the number of nodes that disclose their position. We consider an algorithm for self selection of emitters. During the localization process, each of the new localized node will emit only if it provides the best coverage for an area covered by any pair of its references. Our algorithm eliminates the message overhead of a set of nodes closely situated by selecting the one situated at the best position from the coverage point of view.

Our main results are to show through simualations that our algorithm succeeds in inhibiting all the transmissions which are not necessary for the localisation process to succeed, thus making it (in a weak sense) optimal, and to find a robust waiting time function (used in the inhibition process) which guarantees success of the localisation process in a short running time.

The paper is organized as follows. We begin with a presentation of the current work in localization of sensor networks in section II. Section III describes the details of the proposed algorithm. In section IV we validate our algorithm by the means of experimental results. Finally we present our conclusions in section V.

2 State of the art

The localization process starts up with a small number of known position nodes, called anchors or beamers. They advertise their position in the network and the information is used by the other nodes to compute their own location. Based on the geographical position of the anchors and the estimated distance to them, a node can compute its own location, in a process called atomic multilateration.

Range based algorithms use different physical measurements for estimation of the distance to known location entities. In range based protocols [6],[7], [8] position awareness is propagated through the network during an iterative process. Each node, after receiving at least three known references, computes its location and became a source at the next step. The drawback of this algorithm is the message overhead and the increase probability of collision generated by simultaneous located nodes.

Range free algorithms approach this problem by using different methods to compute location without measuring the distance to the anchor. They use metrics, like for example the hop count to an anchor to estimate the coordinates, making the message overhead even higher. In [9], each anchor broadcasts a message into the network, containing its location. Each node waits for all the beamers and computes its location based on the centroid method that considers all known locations. The traffic overhead depends on the number of sources. DV-hop [10] keeps track of the number of hops to the reference and computes location based on the average distance between one hop neighbors. For each anchor, a message with the coordinates is broadcasted into the network. After the anchors collect information related to the coordinates and hop count of the entire set, they compute and broadcast into the network the average range. Compared with the centroid algorithm, DV-hop will include additional traffic.

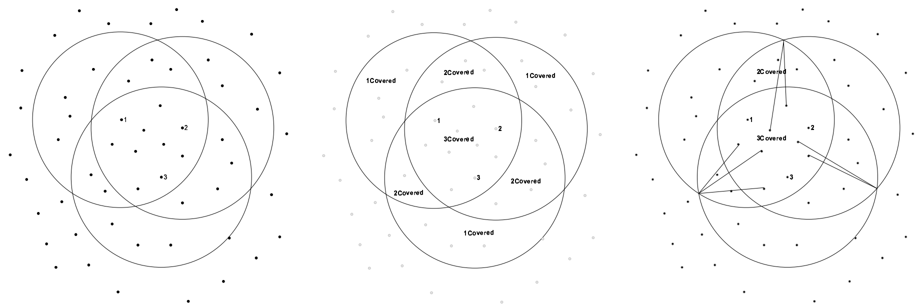
3 Algorithm description

The condition to initialise the algorithm is that 3 anchors nodes are close enough from each other to have common neighbours. The anchors start by broadcasting their position. A node is able to localize itself if it receives at least 3 location advertising messages. In an iterative process, like the greedy algorithm, every newly localized node broadcasts its position. Our aproach is to inhibit the redundant emissions.

Consider the case of three references (localised nodes), each of them in the range of the other two, as shown in Fig. 1(a). The area covered by a set of references is divided in three regions. The *3Covered* area contains the nodes which are in the transmission range of the three references: they will become location aware. The *2Covered* area contains nodes in the transmission range of each pair of references. The *1Covered* area contains nodes in the transmission range of only one reference. The neighborhood of the nodes and the division in several regions is presented in Fig. 1(b).

Each node in the network listens for three location advertising messages. After it receives three messages, it is able to compute its geographical position (by using the *calculate_coordinates()* function).

In order to propagate the location awareness, each *2Covered* area needs a new emitter. The nodes in the *3Covered* area are good candi-



(a) The neighborhood of the initial anchors (b) The regions of the covered area (c) The evaluation of the potential references

Fig. 1.

dates for being this new emitter. Our idea is that a node should send its coordinates only if it is the closest node to one of the *2Covered* area. The distance evaluation is made by the *calculate_distances()* function. Given a newly localized node, we define its *critical point* as the closest point among the intersection points of the tree pairs of coverage circles, not adiacent to the *3Covered* area. The evaluation of the distances to the *critical point* is shown in Fig. 1(c).

A newly localised node sets a timer before deciding if it will broadcast or not, as discused in 3.1. If a node receives a message from a neighbour closer to its critical point before the timer expires, then its transmission is canceled and the timer is stopped and no message is sent, c.f. 3.2.

The algorithm is presented below:

3.1 Emitters inhibition

The main feature of the algorithm is the prevention of the emission of some of the nodes, while still completing the localization process. The purpose of a 3Covered area node emission is to provide a third source to the nodes from a 2Covered area. The inhibition takes place when a better transmission is heard, by dropping the timer and the broadcast. If a node hears a better emitter before its timer expires, it will decide not to emit. The *check_proximity()* function defines if an emitter is better or not.

A node will consider its distance to the critical point in order to take an inhibition decision. The emission inhibition with selection verifies if the emitter is situated closer to the same critical point as the current node, if this is so, it considers the new emitter to be better. An alternative is the emission inhibition without selection, where a node drops the counter if it

Algorithm 1 Emitting decision algorithm

```
wait for 3 anchors
calculate_coordinates()
calculate_distances()
set_timer(distances)
while time_is_active do
    if new_message() then
        check_proximity()
        if closer then
            drop_timer()
            exit()
        end if
    end if
    if end_timer() then
        broadcast_coordinates()
    end if
end while
```

hears a message from any of its neighbours, without previous verification of the proximity.

3.2 Time selection formula

A first approach is to randomly initialize the timer. It reduces the complexity of the algorithm, but better performances can be achived. Indeed, a better criteria is to base the timer initialization on the distance to the critical point. *set_timer(distances)* computes a value based on the distance parameter.

A scale factor is also necessary, in order to normalize the interval of the possible values of the timer. Two characteristics of the algorithm are influenced by the scaling factor. The first is the overall propagation time of the location process. The second is the number of overlapping characteristics. The choice of a scaling factor is thus a tradeoff between the performances measured by the two characteristics. If the scale factor is to large, the propagation delay is also large and if the scaling factor is too small, the number of overlapping transmissions becomes too large.

4 Simulations

In this section we numerically validate the expected behaviour and performances of our algorithm. The simulations we present compare our localisation algorithm and the greedy localisation algorithm which is considered to be the reference. To compare algorithms, the criteria we are intersted in are (a) weather the localisation algorithm succeeds (are sensors localized at the end of the algorithm?) (b) the total number of emmitting sensors (c) the running time of the algorithm. The numerical experiments we present show that our algorithm (emission inhibition) competes with the greedy algorithm in terms of localisation success while significantly reducing the total number of emmiting sensors. Experiments also show that the inhibition *with selection* algorithm has more chances of making the localisation process succeed than the algorithm using inhibition, but *without* selection. In the case where collisions are considered, slowing down the localisation process by making sensors wait a long time before emitting their position is an obvious way of reducing the number of collisions, however this also augments the time required for the localisation process to finish. The simulations we present show that a waiting time which is increasing like the square of the critical distance, see Figure 1(c) seems to be perticularily appropriate and has a slowly increasing running time, which is an unexpected and pleasant property.

4.1 Details on the experiments and the representation of results

An experiment consists in randomly dropping $n = 1000$ sensors in a 1 by 1 square. Three localised sensors are dropped in the middle of the square, and the localisation starts: each sensor with at least three localised neighbours (who emitted their localisation) becomes localised and broadcasts its position to all his neighbours (according to the algorithm which is being tested). The process goes on iteratively. In order to avoid the impact of the border effects we stop the simulation as soon as 500 sensor nodes are localised, and consider the localization process *succesfull*). Each experiment is repeated 2000 times, and the outcomes are presented in a box plot graphic.

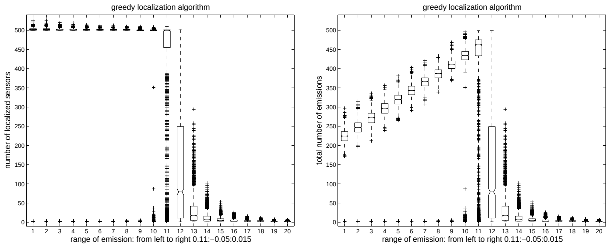
Box plots are composed of a box, the lower line being the lower quartile, the middle one the median and the upper one is the upper quartile of the sample. The median is surrounded by a notch wich shows the 95% confidence interval (for the median). When comparing two medians, they are considered to be signicantly different only if one is not in the confidence interval of the other (i.e. if one median is not in the notch around the other median). The dashed lines extending above and below the box show the span of the other samples. The plus sign represents outliers. Statistically, outliers represent data which is suspected to be unsignificant, perhaps resulting from an input data error or bad measurement. However, in our setting the presence of outliers shows that we are in the transition phase between the regime where the localisation algorithms succeeds and the regime where it fails and the change of regime is not yet significant enough to be included in the statistical box plots.

4.2 Results

We present two sets of experiments. For the first set of experiments, collisions due to simultaneous transmissions are not taken into consideration. We are interested in observing how small the transmission radius can become while still allowing the localisation algorithm to succeed, as well as the total number of emmiting sensors. This set of experiments allows us to validate the inhibition algorithms (with and without selection) as significantly reducing the number of emissions while still competing with the greedy algorithm in terms of succeeding in localisating sensors.

The second set of experiment takes collisions into account. This set of experiments allows to compare the total time needed for the differents algorithms to achieve localisation. Intuitively, when a sensor is localised, it waits a certain time before broadcasting its position. The longer the waiting time, the less collisions their will be (which increases the probability that the localisation algorithm succeeds), but on the other hand, a longer waiting time will make the overall localisation process slower. We investigate the tradeoff between high sucess probability and fast localisation. The waiting time before retransmission is controlled by a parameter k. We find out that setting a waiting time quadratic in the distance to the critical region ensures that the inhibited with selection algorithm runs fast, independently of the scaling parameter k.

Numerical experiments without collisions The first set of experiments is composed of figures 2, 3 and 4. We look at the impact of the transmission range on the performance of the localisation algorithms. The x-axis of every figures corresponds to the range of emission which varies from 0.11 to 0.015 with a step of 0.005. With 1000 sensors in a 1 by 1 square, this corresponds to an expected number of neighbors ranging from 38 to 0.7 with a step of approximatively 2.5. On the left of each figure is the number of localised sensors versus the emission range. Because we stopped the simulations as soon as 500 sensors are localised, having 500 localised sensor means that the localisation algorithm has succeded. As this number deacreses the algorithm fails. On the right side of the Figures, we plot the total number of emission which occured at the time we stopped the simulations versus the emission range.



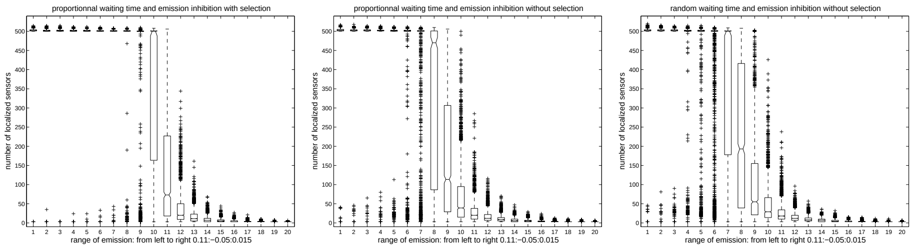
(a) Localized sensors (b) Emitting sensors

Fig. 2. Greedy strategy

Looking at the Figures 2(a) and 3, we observe that the greedy algorithm is the more robust in the sense that it succeeds with a smaller transmission range than our algorithm (figure 3(a): emission inhibition with selection. The performances of our algorithm starts to decrease with an emission range of about 0.07 and the greedy algorithm with 0.05. This is expected, since with the greedy algorithm every localised sensor emits and this behaviour is necessarily more succeessfull than the inhibited algorithms (at least when collisions are not considered). Also, as expected, we observe on the right hand side of these Figures that our algorithm runs with far less emissions than the greedy one (in the case of the largest emission range about 120 emissions against 230). We also notice that for the smallest ranges, when the greedy algorithm is the only one succeding in localising sensors, the number of emissions is very large: nearly all the sensors have to participate to ensure that the localisation process succeeds. In this configuration, our algorithm inhibits some transmissions which were necessary to ensure the success of the localisation process: this explains why our algorithm stops working before the greedy algorithm and it suggests that that no localisation (trilateration) algorithm could make the localisation process succeed while significantly reducing the total number of emmissions (i.e., it would be running an “almost” greedy algorithm). Comparing Figure 3(a) and Figure 3(b) we notice that preventing the emission of sensors in a selective way (a sensor stops its timer only if the received data comes from a sensor located closer to the intersection of the 2 neighbours covarage circles) has an important impact. Indeed, numerical experiments confirm that the selective inhibition avoids to prevent a sensor to emit without the preventing emitting data covering the region it was expected to cover.

Comparing Figure 3(b) and Figure 3(c) confirms that choosing a waiting time increasing with the distance to the critical point of the sensor (c.f. definition 3) results in better performances than having a random waiting time (in terms of seeing the localisation algorithm succed even for small transmission ranges). This comes from the fact that, in the case of the random waiting time algorithm, sensors situated far from the *2Coverage* region (c.f. Figure 1(b)) sometimes emit befor sensor closer to the critical *2Coverage* region. As a consequence, for the random waiting time, some of the sensors which are close to the *2Coverage* region no not emit although this would have permitted the localisation of the neighbours in the *2Coverage* regions. On the other hand, the total number of emission is smaller for the random waiting time algorithm than for the inhibition without selection algorithm because of the larger number of inhibitions. This strategy seems to be adequate as the range of emission is quite large.

As a first conclusion, these experiments show that the *inhibition with selection algorithm* significantly reduces the total number of emissions required to make the localisation process succed and that *it is optimal amongst inhibition algorithms* in the following sense: when the inhibition with selection algorithm starts to fail, the greedy algorithm requires almost every sensor to transmit.We also observe that when the expected number of neighbours (or equivalently when the transmission range or the density of the nodes) increases, the inhibition algorithm *without* selection and the inhibition algorithm with random waiting time, (c.f. Figure 4(b), 4(c)) compete in making the localisation process succeed. Moreover, when they significantly reduce the total number of emissions, even when compared to the inhibition with selection algorithm.



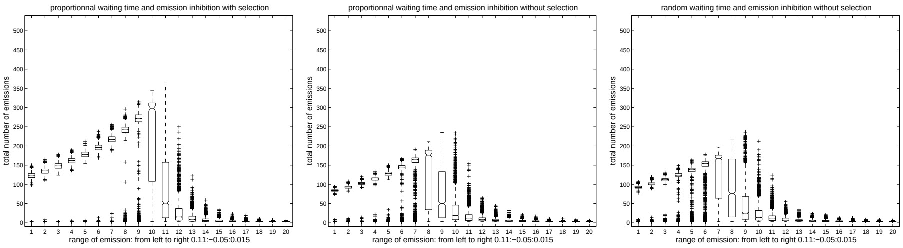
(a) Inhibition with selection

(b) Inhibition without selection and proportional waiting time

(c) Inhibition without selection, random waiting time

Fig. 3. Localized sensors

Numerical experiments with collisions The second set of experiments deals with collisions and the impact of the waiting time function on the total time required for the localisation process to finish. We limit our study to the inhibition with selection localisation algorithm and compare it to the greedy algorithm. Also, we fix the transmission radius to



(a) Inhibition with selection (b) Inhibition without selection and proportional waiting time (c) Inhibition without selection and random waiting time

Fig. 4. Emitting sensors

r=0.11. We compare two waiting time functions,

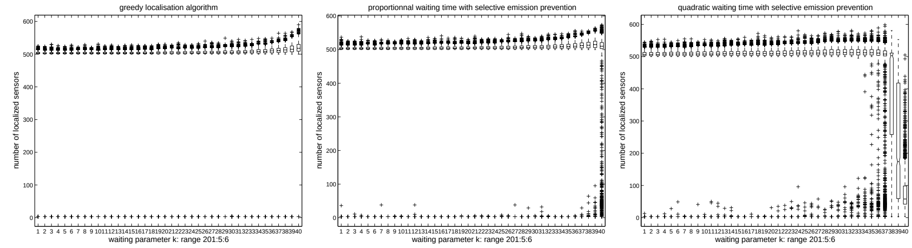
1) Poportionnal waiting time $f(k) = kd$,
2) Quadratic waiting time $f(k) = kd^2$.

We consider a normalised distance d which is the distance of a sensor to its critical intersection point divided by the transmission range of sensors, and we introduce a normalized parameter k.When a sensor is localised, it waits for a time $f(k)$ and broadcast its position (unless it becomes inhibited by a transmission from another sensor). Collisions are modeled in the following way: time is divided in discrete rounds, and if a sensor receives more than one message in the same round, it drops all of them . Intuitively a quadratic waiting time seems more appropriate since it is proportional to the area covered by the region and hence, on the expected number of sensors in the region.

Numerical results are plotted on Figures 5, 6, on the left side of the figures is plotted the number of localised sensors versus the parameter k which ranges from 201 to 6 with step 5. On the second plot of each of the Figures, we represented the total number of time before the simulation stops (either because 500 sensors are localised and the algorithm has succeeded, or because the localisation process fails). Notice that in the three Figures it can be seen that the total number of localized sensors often exceeds 500. This follows from the short waiting time which implies many simultaneous emission and the way the timeout criteria is implemented in our simulations.

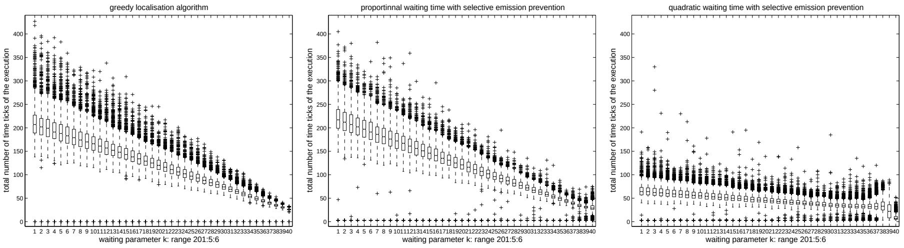
Since the greedy algorithm implies that more sensor emit their position (c.f. section 4.2), it could have been that collisions have more impact on the greedy algorithm that the inhibition with selection algorithm.

However, in terms of success of the localisation process, simulations shows that this is not the case: the greedy algorithm is again the more robust and numerical results show that it works in localizing sensors even when the waiting time parameter k becomes smaller.



(a) Greedy (b) Proportional waiting time (c) Quadratic waiting time

Fig. 5. Localized sensors



(a) Greedy (b) Proportional waiting time (c) Quadratic waiting time

Fig. 6. Execution time

On the contrary, the quadratic waiting time function makes the inhibition with selection localisation algorithm fail for the smallest values of k (figure 5(c)), a patological behaviour which is not as important when a proportional waiting time is used (c.f figure 5(b)). However, the important factor (once the localisation process succeds) is not to minimize k, but rather to minimize the *total running time of the localisation process.* With respect to this objectives (successful and fast localisation), simulations are to be interpreted as showing that the quadratic waiting time is the best.

Indeed, the running times of the greedy algorithm and the algorithm with proportionnal waiting time function are very similar (compare Figures 6(a) and Figure 6(b)) and even for small values of k, these running time are significantly greater than the running time of the algorithm with the quadratic waiting time for the choice of a greater k (c.f. figure 6(c)). Moreover, an important characteristic of the results obtained with the quadratic waiting time function is that the total running time is less sensitive to the value of the parameter k than the two others algorithms.

5 Conclusion

The numerical experiments conducted and presented in this paper show that the scheduling strategy introduced in this paper combined with emission inhibition with selection is good in minimizing the total number of emission as well as in reducing the total running time of the algorithm. Possible applications of this strategy would be to reduce the energy consumption by reducing the number of emissions, reducing privacy disclosure in the case security is a concern and reducing the total running time of the localisation phase. Intuitively, the strength of this algorithm might increase with the density of the sensors since the emission inhibition becomes more efficient. A possible future research direction would be to investigate theoretically the behaviour of our algorithm, including the impact of density.

References

1. Luo, H., Ye, F., Cheng, J., Lu, S., Zhang, L.: Ttdd: A two-tier data dissemination model for large-scale wireless sensor networks (2003)
2. Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed diffusion: a scalable and robust communication paradigm for sensor networks. In: Mobile Computing and Networking. (2000) 56–67
3. Powell, O., Leone, P., Rolim, J.: Energy optimal data propagation in wireless sensor networks. arXiv.org automated e-print archives (2005) Report CS-0508052, journal version submitted for publication.
4. Efthymiou, C., Nikoletseas, S., Rolim, J.: Energy balanced data propagation in wireless sensor networks. (Invited paper in the Wireless Networks (WINET, Kluwer Academic Publishers) Journal, Special Issue on "Best papers of the 4th Workshop on Algorithms for Wireless, Mobile, Ad Hoc and Sensor Networks (WMAN 2004)")
5. Leone, P., Nikoletseas, S., Rolim, J.: An adaptive blind algorithm for energy balanced data propagation in wireless sensor networks. In: The First International Conference on Distributed Computing in Sensor Systems (DCOSS). Number 3560 in Lecture Notes in Computer Science, Springer Verlag (2005)
6. Savvides, A., Han, C.C., Strivastava, M.B.: Dynamic fine-grained localization in ad-hoc networks of sensors. In: Mobile Computing and Networking. (2001) 166–179
7. Savarese, C., Rabay, J., Langendoen, K.: Robust positioning algorithms for distributed ad-hoc wireless sensor networks usenix technical annual conference (2002)
8. Aspnes, J., Eren, T., Goldenberg, D.K., Morse, A.S., Whiteley, W., Yang, Y.R., Anderson, B.D.O., Belhumeur, P.N.: A theory of network localization. Unpublished manuscript. To appear, IEEE Transactions on Mobile Computing (2006)
9. Bulusu, N., Heidemann, J., Estrin, D.: Gps-less low cost outdoor localization for very small devices (2000)
10. Niculescu, D., Nath, B.: Ad hoc positioning system (aps) using aoa (2003)